A Game Loop Architecture for the Modeling and Simulation of Mission Threads

Tom Tanner
thomas.l.tanner@saic.com
540 663-4867 (Office)
540 903-6713 (Cell)

Outline

- Objective
  - To promote the viability of the game loop architecture as a means of modeling mission threads
  - Required elements of proof
    - Organizational relevance
      - Impetus for promoting acceptance, assimilation and socialization
      - Preferably a key role within a current strategy, initiative or movement
    - Added value
    - Achievable and affordable
- Presentation
  - Alignment with Systems 2020
  - Mission Threads - Mission Thread Models
    - Brief (localized) description
    - Their role in the mission systems engineering process (the analyst's perspective)
  - The Game Loop Architecture
    - Mission centric vs. system centric approach
    - Components
    - Game loops
  - Example
    - "Swarm Raid" mission thread from CGX program
    - Detailing the game loop
  - Transition from wireframe to system design
  - Summary and Recommendations

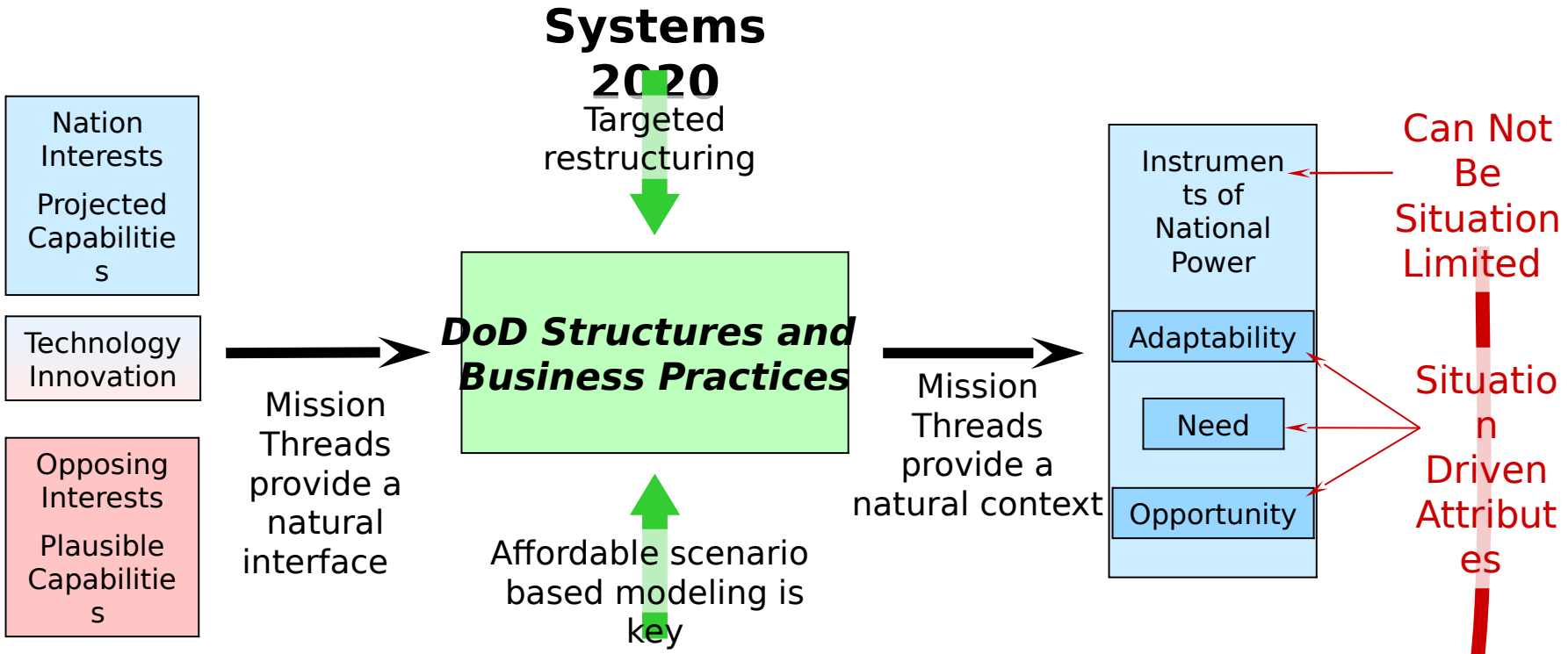
Mission Threads in Support of Systems 2020

- Closing The “Churchillian Delta”
  - Looking back at the lessons of WWII Sir Winston Churchill wrote:

    “It's not enough that we do our best; sometimes we have to do what's required.”
  - Initiatives that matter target the delta between “our best” and “what’s required”
  - In a plenary address at the NDIA Systems Engineering Symposium Kristen Baldwin framed the Churchillian Delta that is driving Systems 2020 as

    “DoD structures and business practices are not structured for adaptability, need or opportunity”
- Situational Dependence
  - Adaptation is a response to situation
  - Need and opportunity are both created by situation
  - Mission Threads provide a useful mechanism for establishing situational context

Setting Situational Context With Mission Threads



Situations are likely to be "crafted" by the opposing interests

Mission Scenarios must be an integral part of the process.
Ideally they should span the limits of plausibility.
Collectively they must demand a broad range of capabilities.
Mission scenarios must be played out at the enterprise

The Game Loop approach brings economy of scale to the modeling, simulation and analysis of mission

Localizing Systems 2020 Goals

Executive Summary

Findings

- Significant opportunities exist to develop and deploy technologies to strengthen the Department's ability to conduct rapid capability fielding
  - However, non-technical challenges (e.g. cultural, budgetary, contracting, etc) must be simultaneously addressed
- ⇨ Greatest leverage in the "front end" of the life cycle
  - Concept Engineering: Rapidly elucidating the need, exploring solutions, developing CONOPs, and deriving requirements for materiel solutions
    - Virtual environments and rapid physical prototyping are linchpin technologies
- Opportunities exist to increase design, test, and production efficiencies
- ⇨ – Examples include physics-based M&S to reduce testing and model-based engineering and manufacturing approaches

Recommendations

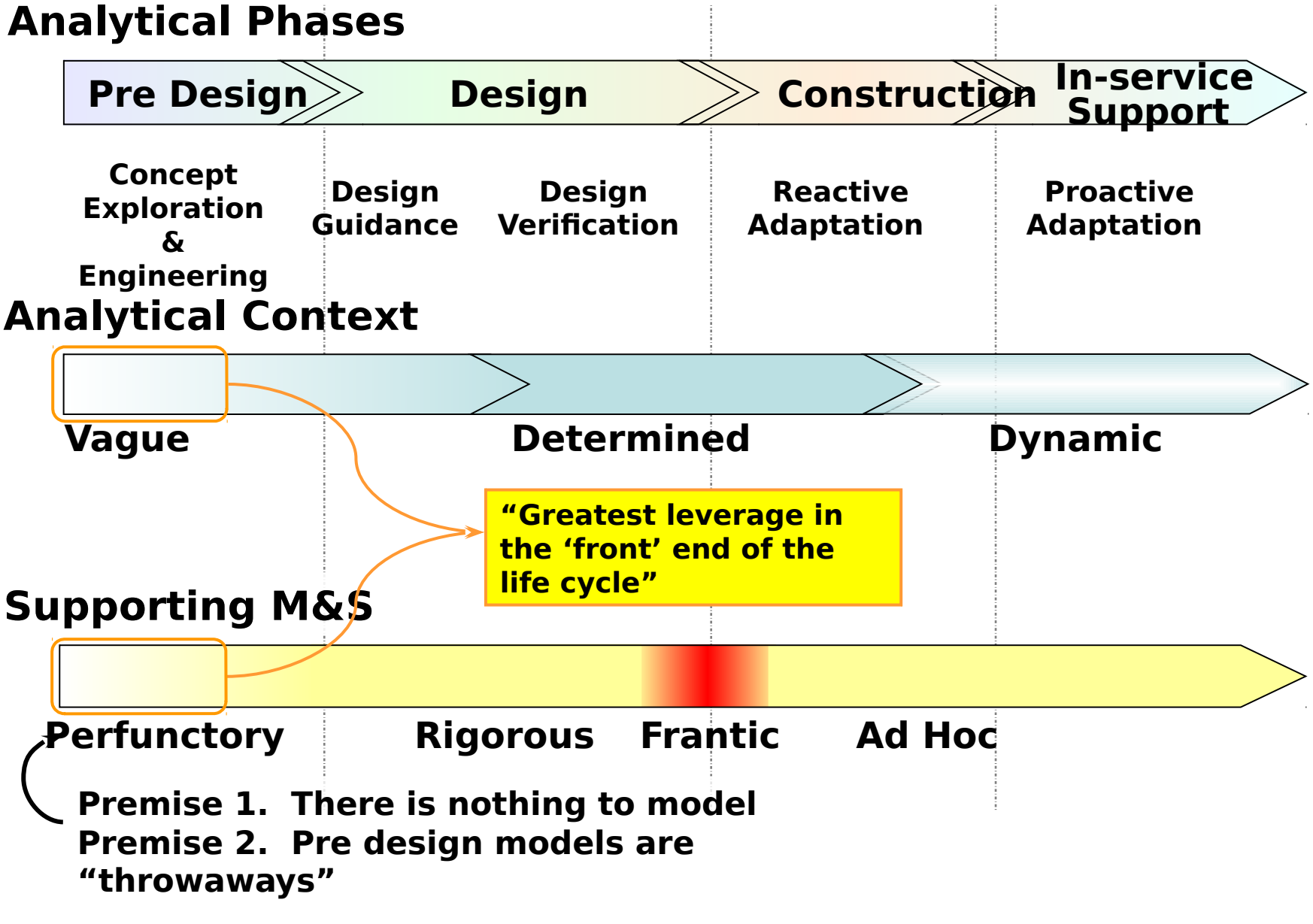
- ⇨ A concept engineering center should be implemented immediately that leverages the substantial existing capabilities across the Department
- A strategic R&D roadmap should be developed and implemented to mature and transition emerging tools and promising innovative ideas
- A set of potential pilots is recommended to demonstrate the application of today's toolset to relevant rapid capability challenges

Rapid Capability Toolbox Study Final Report , March 2010
(Cited as Key Study in Systems 2020)

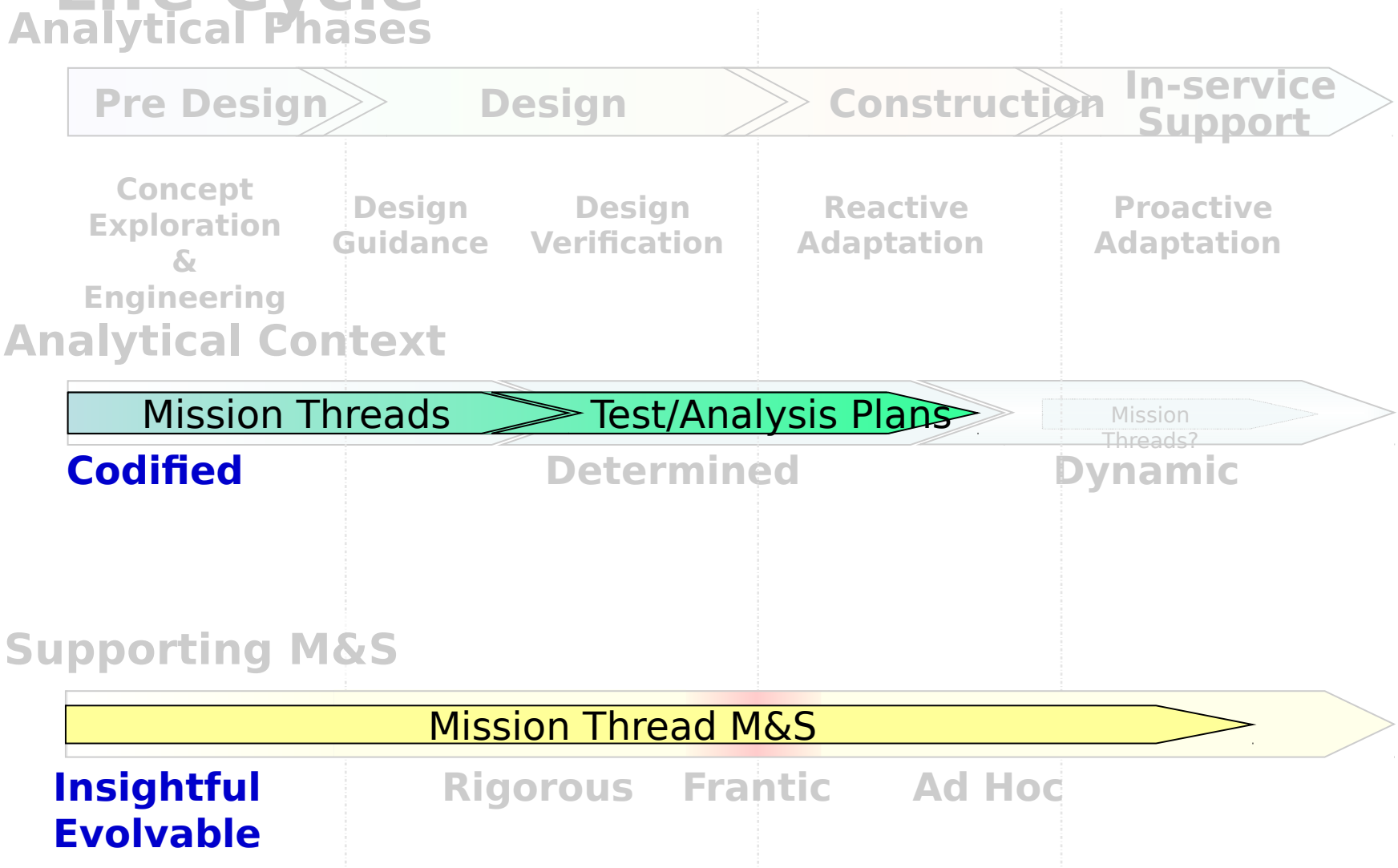
Mission Threads (Quick Look)

- Analytical Context
  - In this discussion “analysis” refers to the assimilation of information into a body of knowledge that will serve as the basis for decisions
  - Analytical Context is the basis and the principle mechanism of the assimilation process
- Scenario Based Context Provides:
  - A roadmap for mutually insightful discussions between operators and designers
  - The basis for assessing the mission ➡ system ➡ causation chain performances ➡ attributes ➡ capabilities ➡ effects ➡ objectives
- Mission Threads (as a type of scenario based context)
  - Qualitative, sequential representation of a mission scenario
  - A rapid means of framing the mission from end to end
  - Typically have prescribed branching decisions to ensure a productive trajectory
  - Ideal infusion point for “what ifs”
  - Well suited to requirements definition and design guidance
  - Useful as precursors of test and analysis plans
  - Current efforts are aimed at making them more quantitative
  - **Agility and rapidity of development do not transfer to M&S**

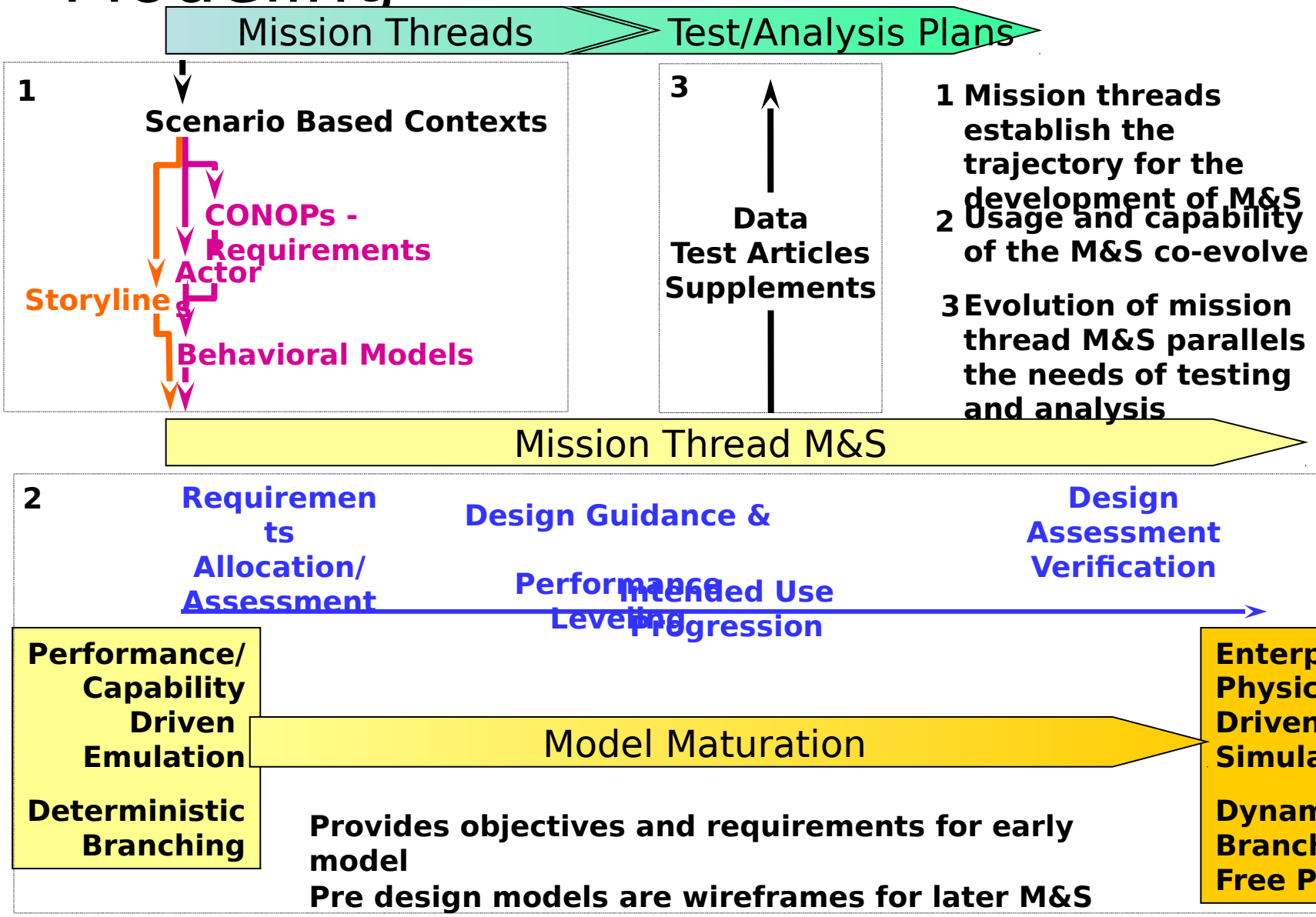
Analytic Life Cycle



Mission Threads in the Analytic Life Cycle



Dynamics of Mission Thread Modeling



Mission Thread Modeling

- Payoffs
  - Requirements bridge between analytical needs and M&S capabilities
  - Scoping mechanism for self contained, piecewise analysis of a complex mission space
  - Continuity of analysis, design and test cases
  - Provides a basis for confidence based on an understanding what the M&S is doing
- Making it Work
  - Experience suggests incremental development starting with a wire frame of essential features using place holders if needed
  - Simulation must be modular, agile, expandable and readily evolvable
  - Host structure must support large scale composability of an eclectic assortment of component models
  - Must be functional at the front end of a project

Two Approaches to Mission System Modeling

System Centric
Model the system and wrap the war around it

Logical appeal in that it leverages the representation of the system of interest

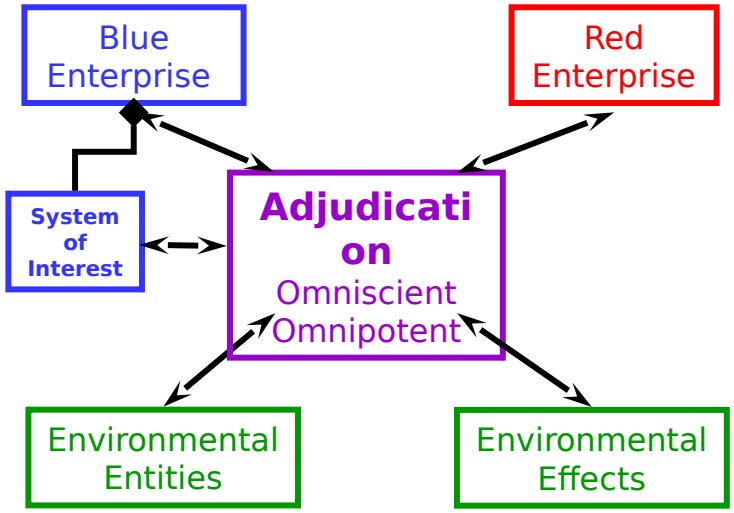
Ideally suited to design assessment

Simulation lags design – limiting usefulness for design guidance

Invites bias in free play scenarios

Decentralized (inconsistent?) adjudication

Mission Centric
Model the war and put the system in it



Simulation can run prior to system design (using place holder if required)

Consistent with the charter of the suggested "Concept Engineering Center"

Mitigates ripple effect induced by changes to the representation of the system of interest

Local adjudication

Context for Game Loops

Game Loop Architecture

- Players
  - Mutually incompatible objectives
  - Capabilities that can influence the end point of the “game”
  - Represented by quasi autonomous objects
- Rule Set and Adjudication Mechanism
  - Rule Set can be combination of convention and “physics” based
  - Centralized adjudication component
    - Fields adjudication requests from players
    - Promulgates results to effected players
- Venue
  - Common Operational Environment
  - Optional “game board”
- Game Driver
  - Consistent with quasi autonomous objec
  - Centralized – Event Queue enforces temporal causality (event drive with time driven and agent based tendencies)
  - Decentralized – Individual Game Loop

The underlying goal is to design mechanisms that make the most of **one time investments** in both the core and in the problem domain.

This combination is a key contributor to “agile, expandable and readily

Legend
Problem Domain
Core

Basic Game Loop Model

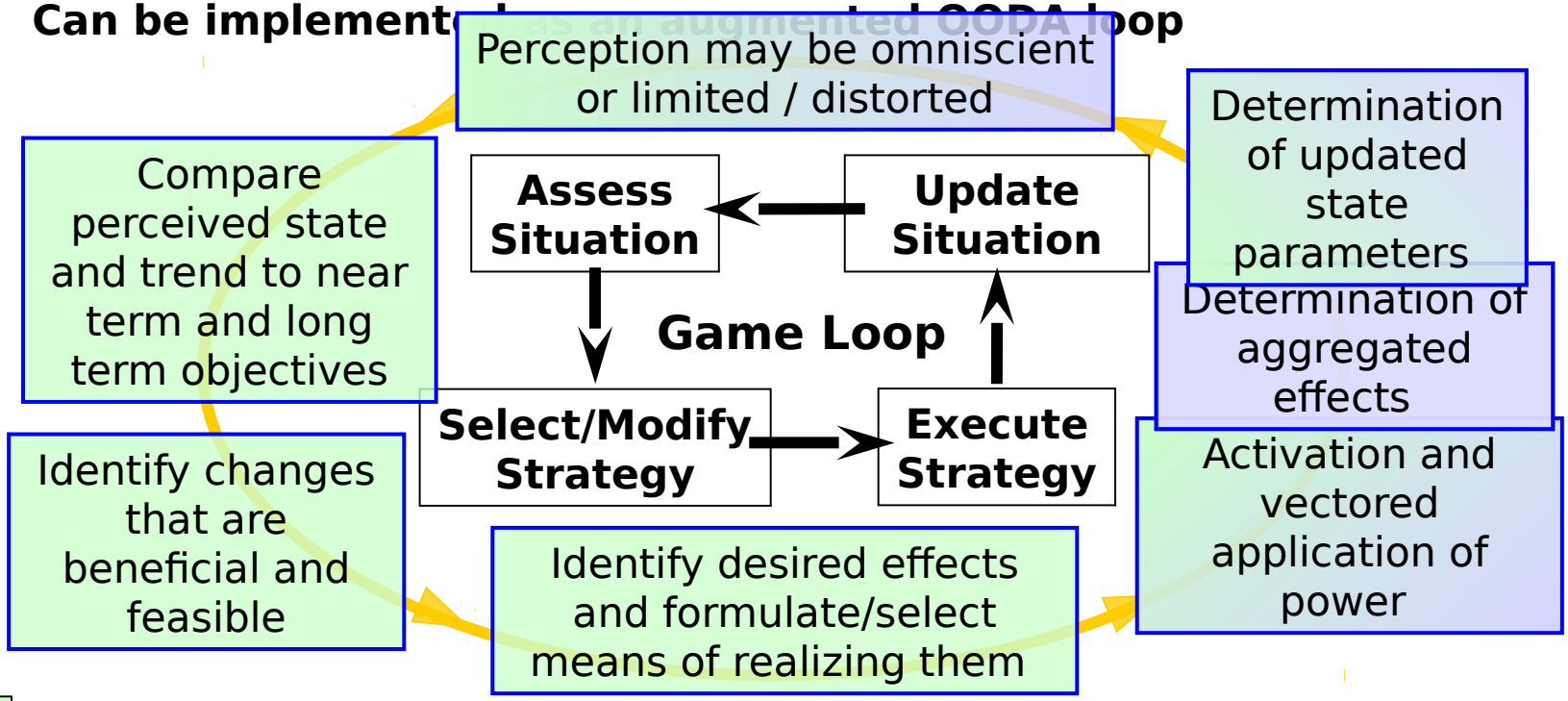
The basic game loop model can be applied any game

Sport, Card, Board, Wargame, Combat …

One loop per player (players of similar type can share the loop implementation)

Situations are based on a shared Common Operation Environment (COE)

Can be implement… as an augmented OODA loop



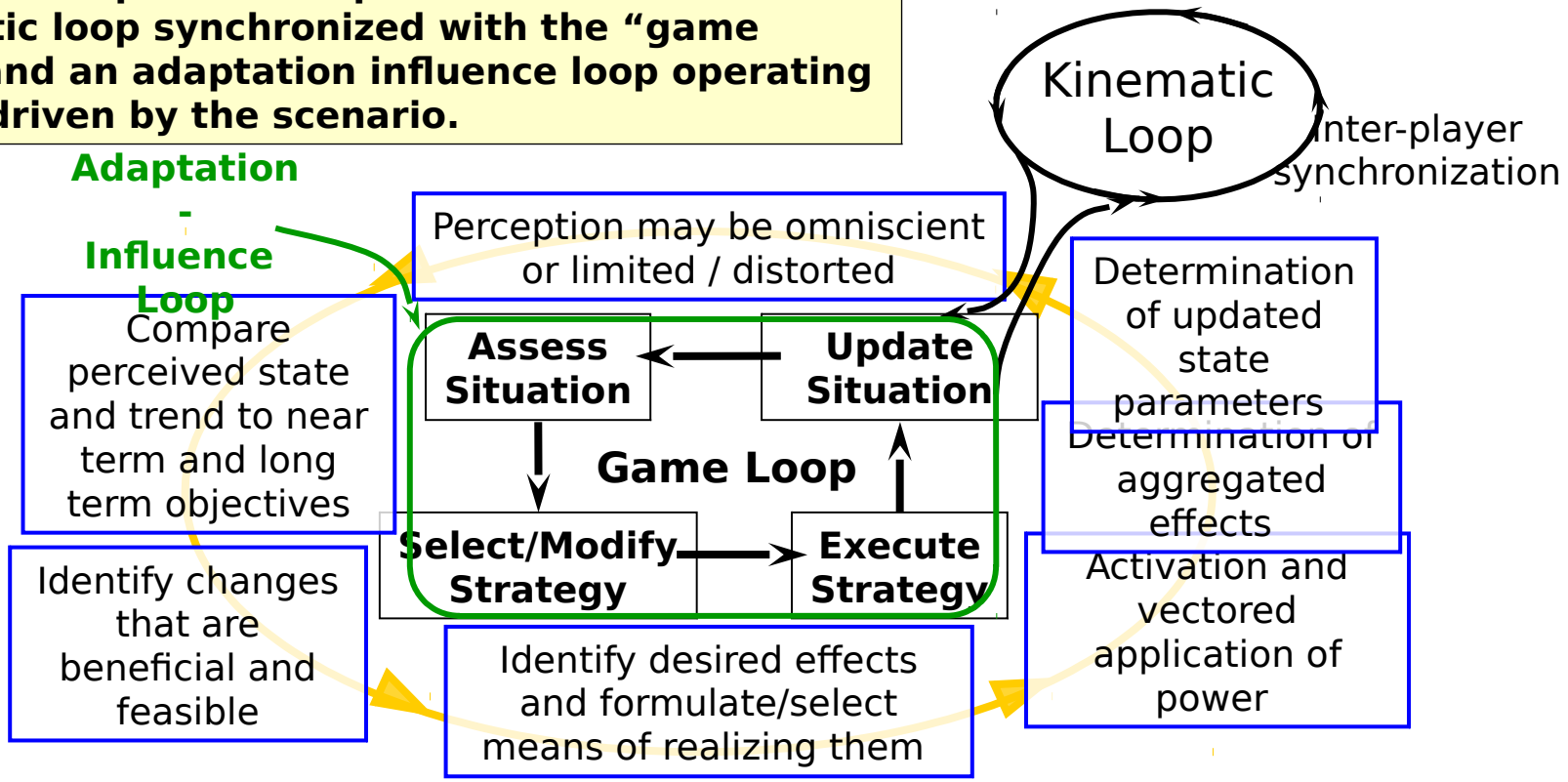
Assigned to Player/Entity

Assigned to Adjudicator

The Modified Game Loop

In simulations of mission systems the speed of the reaction, adaptation process has tactical significance.

The game loop can be implemented as a kinematic loop synchronized with the "game clock" and an adaptation influence loop operating a pace driven by the scenario.



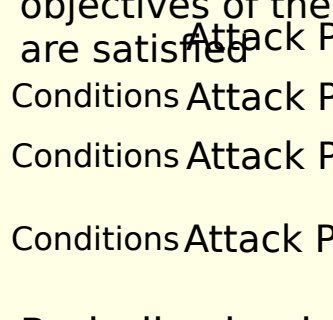
Conducting the Game

Set Up Each player's starting state (position, various load-outs …) is defined in a file.

At least one player or group of players will be assigned an attack profile.

An attack profile may have several phases with conditions and objectives.

A player or group may advance to the next phase when the objectives of the current phase and the conditions of the next phase are satisfied



A given attack phase can be implemented once and used in may different mission threads.

Each attack phase knows what parameters define its conditions and objectives

Threshold values for conditions and objectives are specified in the setup of each mission thread

Periodic checks of objectives and conditions are part of each players game loop

Run Each player acts as directed by its own game loop, pursuant to its assigned role in the mission (may be prescribed by attack profile, may be to loiter on station).

Adjudicator has a list of all players and can access all pertinent information.

Players assess the situation by telling the adjudicator where and how they are looking. The adjudicator then responds telling them what they

Modeling a mission thread is a matter of initializing each of the players, including roles and objectives to varying degrees; and hitting run.
(May require the creation of new player types or attack

Example (Pilot Project)
Swarm Raid Mission Thread from CGX

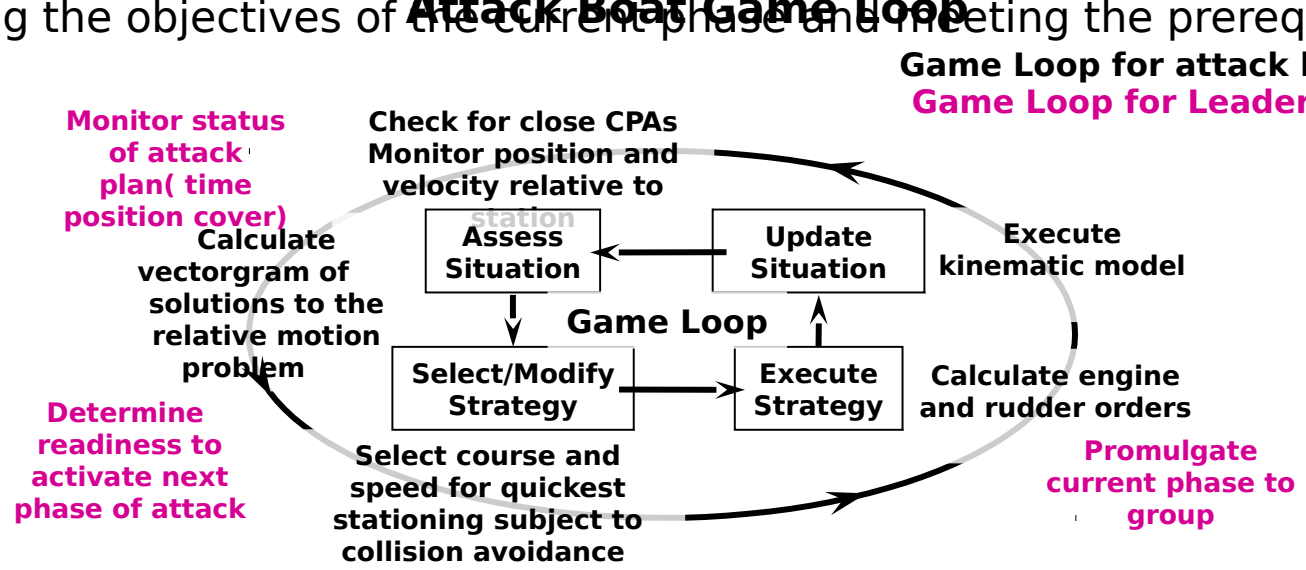
This scenario pits CGX against 15 small, agile, missile capable surface craft in 3 groups of 5

There is a vibrant fishing industry and a shipping lane off shore that will provide cover for the attack. This demonstration is tuned to finding exploitable indicators of a pending attack.

This single scenario can be used to explore and assess all options in the detect to engagement sequence.

Each entity (CGX, small boat, merchant, fishing boat, helo, UAV) executes its own game loop.

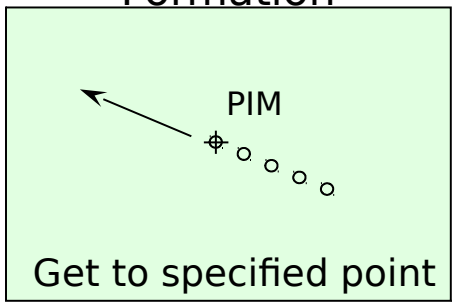
The attack profile consists of 6 phases for each of the groups. Each group has a designated leader that will decided when to proceed to the next phase based on achieving the objectives of the current phase and meeting the prerequisites of the next



Swarm Raid Mission Phases

Transit

Formation

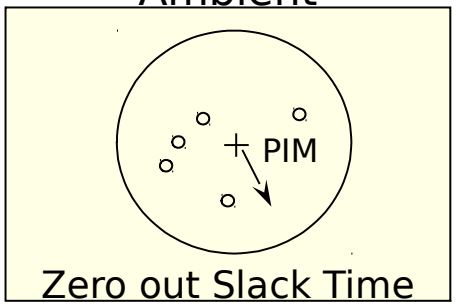


Initial condition - None

Terminal condition - Posit

Loiter

Ambient

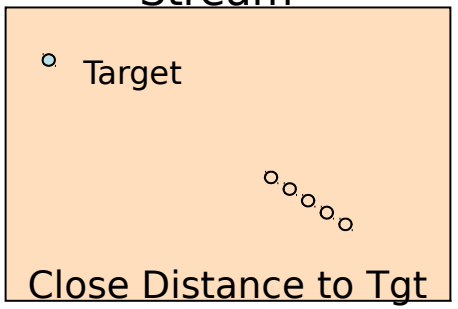


Initial condition - None

Terminal condition - Time or posit

Dash

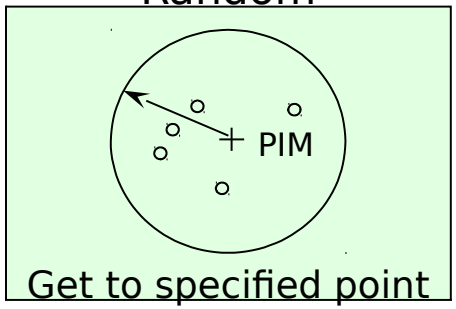
Stream



Initial condition – Time & posit

Terminal condition - Time & posit

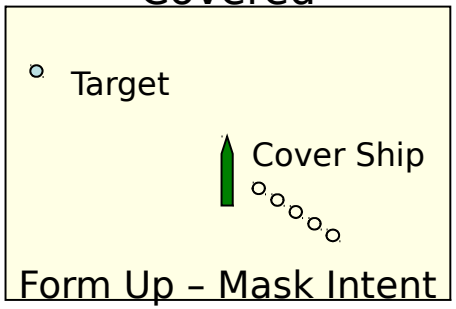
Random



Initial condition - None

Terminal condition - Posit

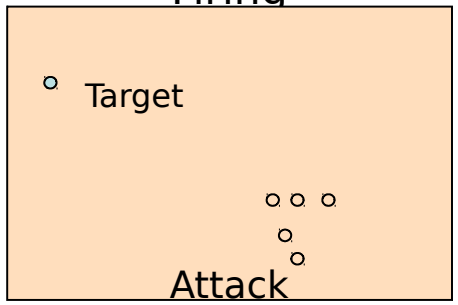
Covered



Initial condition – Target & cover assigned

Terminal condition - Time or posit

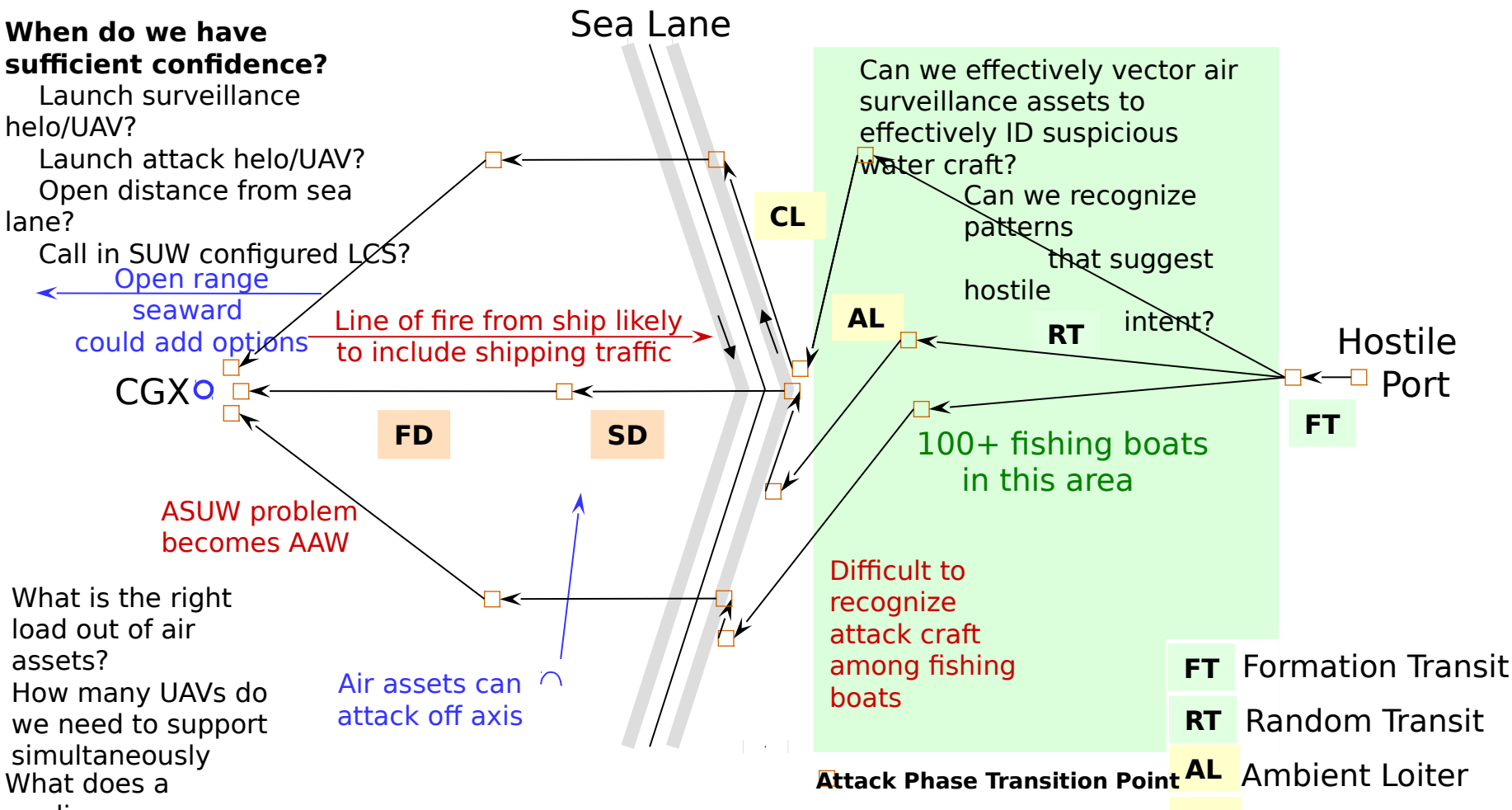
Firing



Initial condition – At launch range (time opt.)

Terminal condition - None

Swarm Raid Scenario Setup

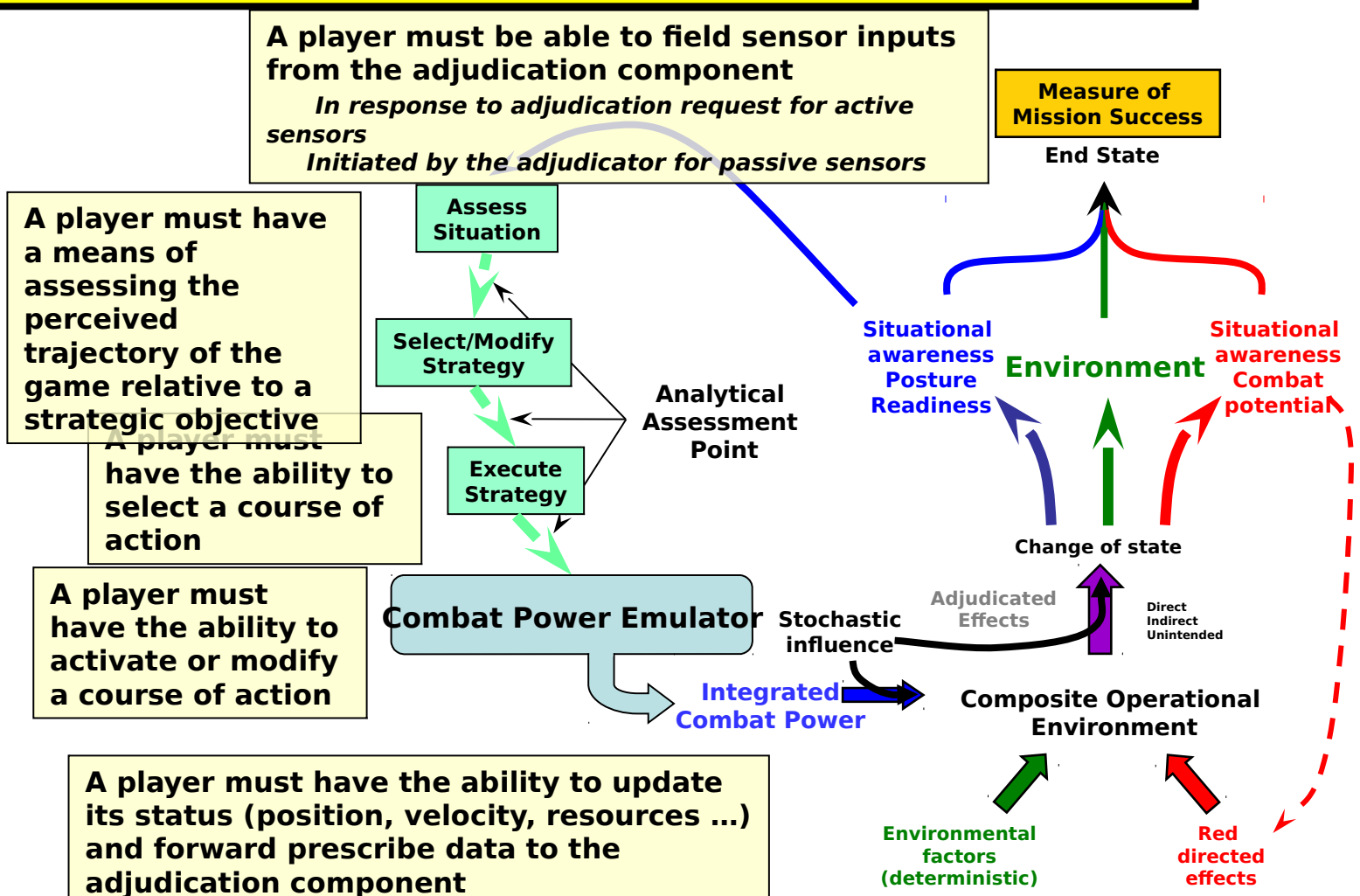


What are we willing to do to operate in this environment? What are the design options for CGX? Other options?
Mission Thread Modeling promotes valuable questions.

Hosting the System of Interest

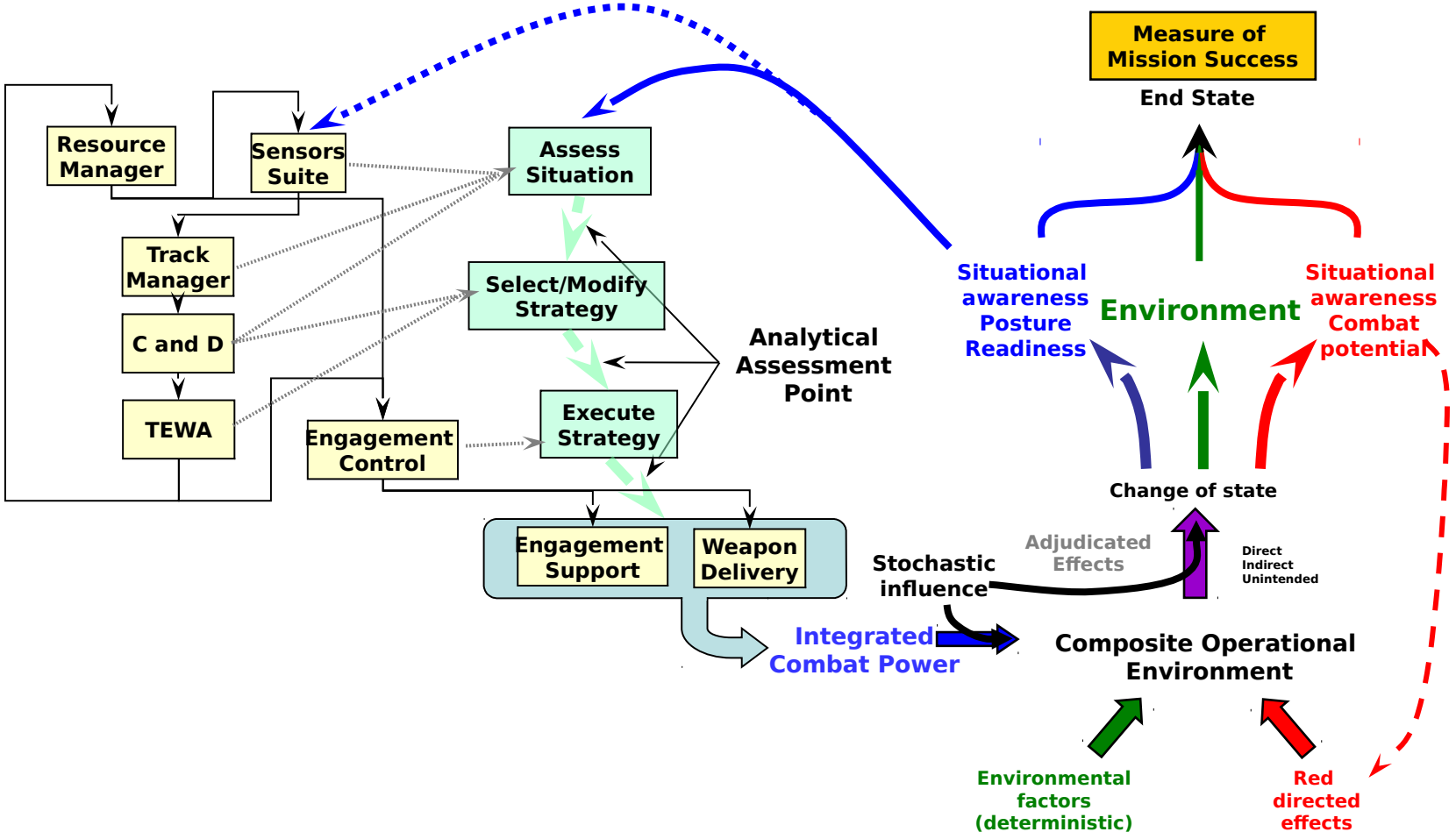
(Closing the Game Loop – integration with the host)

Placeholders are useful for "closing the game loop" providing a complete but rudimentary set of capabilities player must have to participate in the game



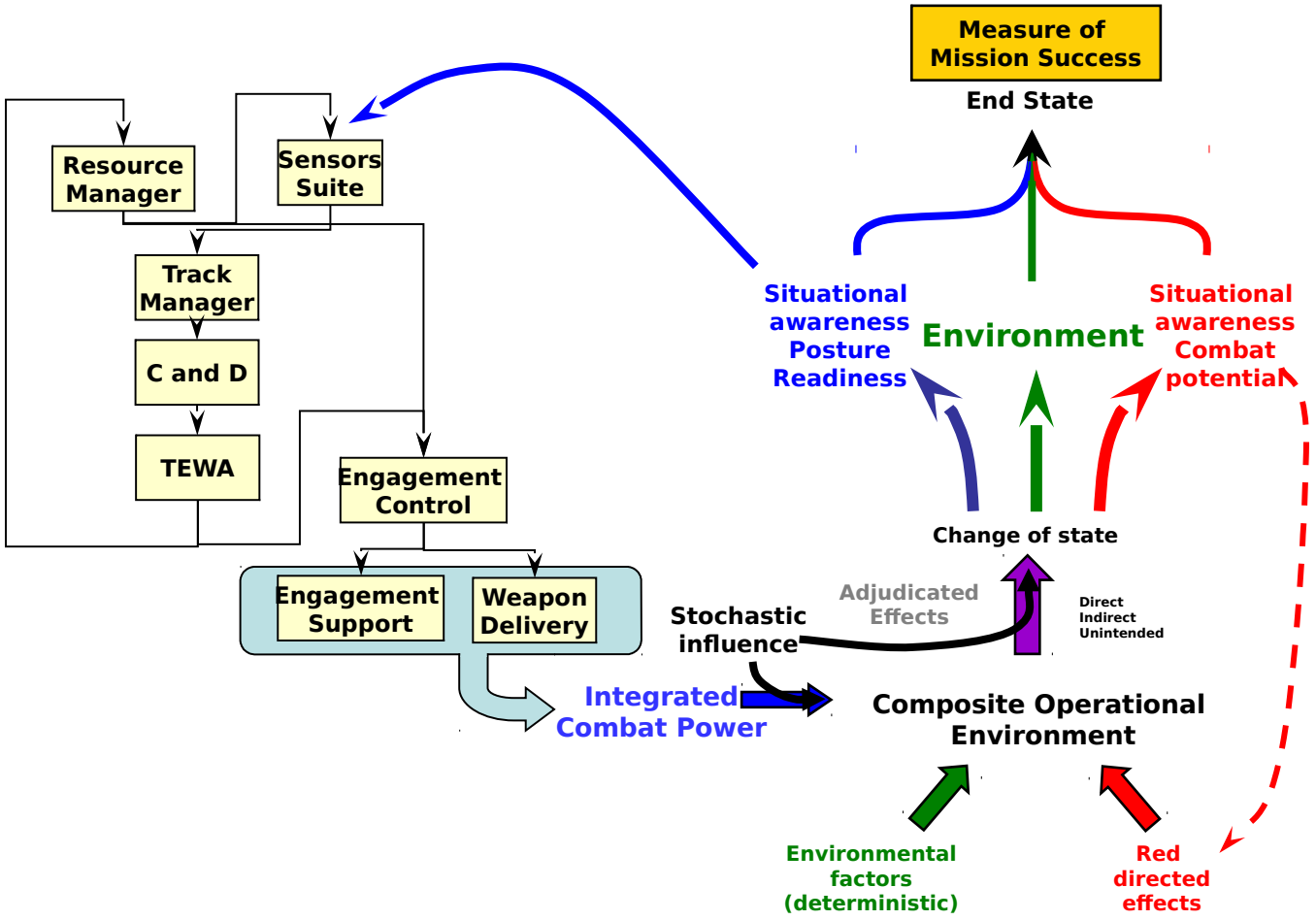
Hosting the System of Interest
(Interim Hosting – The Apprenticeship Phase)

Placeholders can be modified incrementally to incorporate prototypes or equivalent representations of proposed system elements



Hosting the System of Interest
(Functional Hosting)

When the system is sufficiently mature and complete to satisfies the basic game loop requirements the original wireframe can be retired and the system refined incrementally



Summary and Recommendations

- Summary
  - Mission threads are effective and affordable analysis drivers in support of mission system engineering
  - Their utility as simulation drivers is an appealing but elusive goal
  - Mission threads and mission modeling are inline with and are potential enablers of the vitally important “Systems 2020” initiative
  - The game loop architecture is well suited to the needs and the objectives of mission thread modeling and simulation
  - The effectiveness of game loop approach grows with the breadth of the reuse library of entities, mission phases and associated constructs
- Opinion
  - We need the “*Concept Engineering Center*” recommended by the “*Rapid Capability Toolbox Study*”
  - Mission threads and mission thread modeling can provide structure and guidance during the *Concept Engineering* phase
- Recommendation
  - Develop a small nucleolus of expertise in the area of mission thread modeling by way of a suitable pilot program
  - Refine the concept of mission thread modeling
  - Establish a basis of understanding to generate a productive set of standards for the components and structures with which to

Questions

Tom Tanner
thomas.l.tanner@saic.com
540 663 4867 (Office)
540 903 6713 (Cell)

2011 System of Systems Engineering Collaborators Information Exchange Webinars

- ✓ April 12th: **A Game Loop Architecture for the Modeling and Simulation of Mission Threats**, Thomas Tanner, SAIC
- May 3rd: **Mission Engineering for Warfighting Integration of Net-Centric Systems**, Eileen Bjorkman and Timothy Menke, USAF
- May 10th: **The Role of Enterprise Architecture Updates in Guiding Decentralized Organizations**, John Schatz, SPEC Innovations
- May 24th: **Test and Evaluation Issues for Systems of Systems: Sleepless Nights to Sominex**, Dr. Beth Wilson, Raytheon & Dr. Judith Dahmann, MITRE
- June 14th: **Evaluating the Readiness of Federations-of-Models for Use in Simulation-Based Concept Development of Advanced Warfighting Capabilities**

For information, email dasd-se@osd.mil or visit our website: